云存储专家200分钟视频讲解

掌握云存储理论，动手搭建分布式对象存储架构

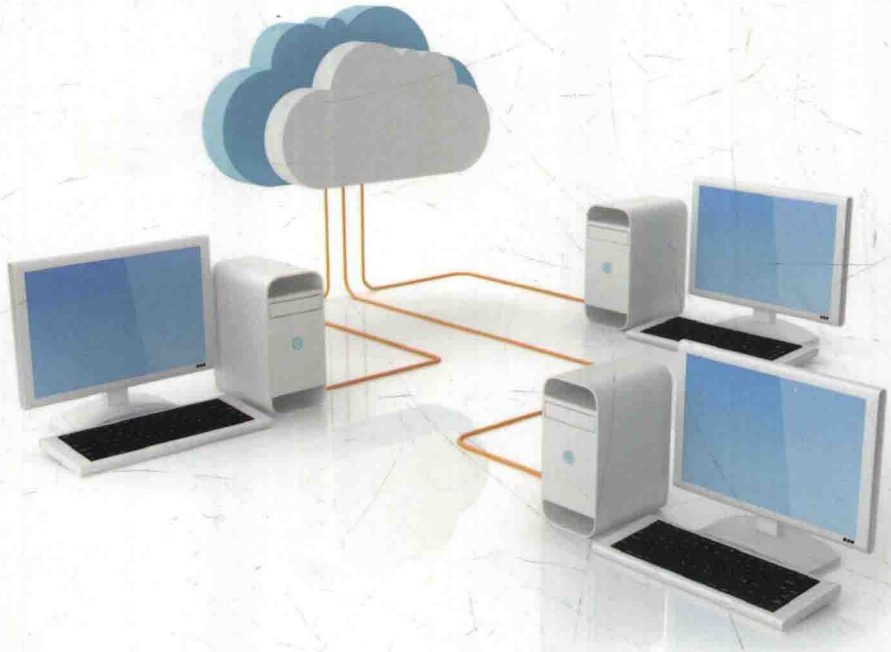
卷积 ARBook

异步图书 www.epubit.com

分布式对象存储

——原理、架构及Go语言实现

胡世杰 著

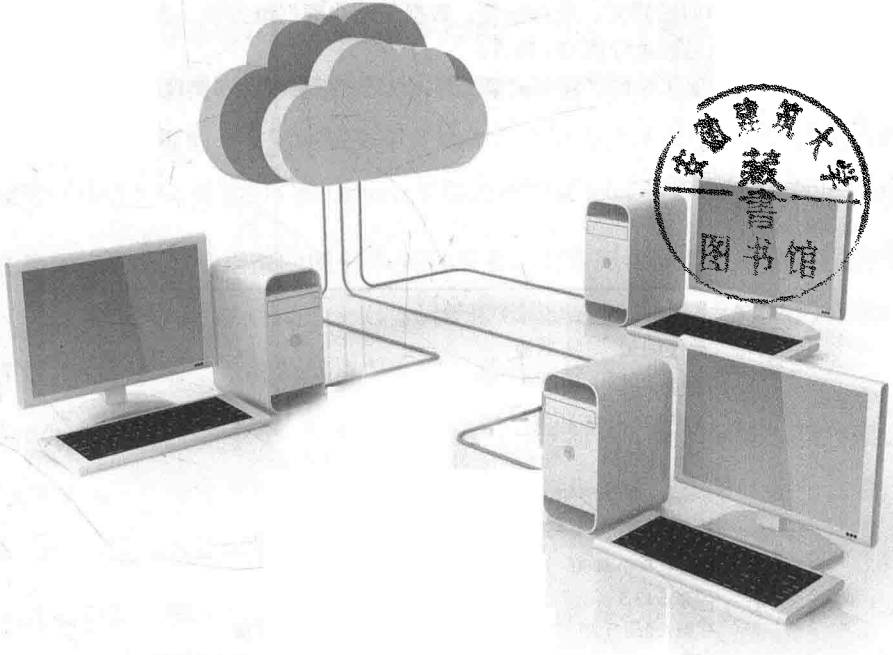


中国工信出版集团

人民邮电出版社 POSTS & TELECOM PRESS

分布式对象存储
——原理、架构及Go语言实现

胡世杰 著



人 民 邮 电 出 版 社
北 京

图书在版编目（CIP）数据

分布式对象存储 ：原理、架构及Go语言实现 / 胡世杰著. -- 北京 ：人民邮电出版社，2018.6
ISBN 978-7-115-48055-2

Ⅰ. ①分… Ⅱ. ①胡… Ⅲ. ①数字信息－存储 Ⅳ. ①G202

中国版本图书馆CIP数据核字(2018)第073119号

内 容 提 要

本书从云存储的需求出发讲述对象存储的原理，循序渐进地建立起一个分布式对象存储的架构，并且将软件实现出来。全书共 8 章，分别涉及对象存储简介、可扩展的分布式系统、元数据服务、数据校验和去重、数据冗余和即时修复、断点续传、数据压缩和数据维护等。本书选择用来实现分布式对象存储软件的编程语言是当前流行的 Go 语言。

本书适合从事云存储方面工作的工程师或架构师，也适合想要学习和实现分布式对象存储的读者。

◆ 著　　胡世杰
责任编辑　陈冀康
责任印制　焦志炜

◆ 人民邮电出版社出版发行　　北京市丰台区成寿寺路 11 号
邮编　100164　　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
固安县铭成印刷有限公司印刷

◆ 开本：800×1000　1/16
印张：13.5
字数：268 千字　　2018 年 6 月第 1 版
印数：1－2 400 册　　2018 年 6 月河北第 1 次印刷

定价：59.00 元

读者服务热线：(010)81055410　印装质量热线：(010)81055316
反盗版热线：(010)81055315
广告经营许可证：京东工商广登字 20170147 号

前言

本书写作目的

早在几年前，云存储还只是存在于业界大佬们口中的一个概念，其应用场景仅供大公司使用。突飞猛进的网络技术似乎在一瞬间就把这个概念普及到千家万户，现在云存储已经是大家司空见惯的一个网络服务了。比如大家用的百度云盘、已经成为实质上的业界标准的亚马逊 S3、微软的 OneDrive、苹果公司的 iCloud 和谷歌的 Google Cloud 等。

现在市面上和云存储相关的图书本来就不多，而专门讲述对象存储实现的书就更是难得一见，且内容大多以 OpenStack、Swift 和 Ceph 这些已经较为成熟的开源软件的架构和实现为例。读者一开始就知道怎么做，然后解释为什么要这么做。

本书则另辟蹊径，完全从云存储的需求出发讲述对象存储的原理，循序渐进、从无到有地建立起一个分布式对象存储的架构，并且将软件实现出来。换句话说，本书首先介绍为什么要这么做，然后解释怎么做。

本书选择用来实现分布式对象存储软件的编程语言是 Go，但并不是非它不可的。读者也可以在了解对象存储的原理之后选用其他的语言来实现。

在读完本书之后，每一位读者都将对对象存储服务这一概念有一个较为深入的理解，部分读者甚至能够实现自己的对象存储服务。

对象存储和云存储的关系

对象存储是云存储的一部分，它提供了云存储后端的存储服务。云存储是建立在对象存储之上的一个整体的解决方案，除了后端的存储服务之外，它还需要包括各种操作系统和平台上运行的客户端、身份认证、多种管理和监控功能等。

本书主要集中在对象存储的原理架构和实现上，对云存储其他组件也会有一定的介绍，但不会是本书的主要内容。

分布式存储的好处

传统的高端服务器性能强劲、成本高昂，以前只有大公司用来搭建自己的私有存储。互联网生态下的云存储则用数量弥补质量，以大量低成本的普通 PC 服务器组成网络集群来提供服务。相比传统的高端服务器来说，同样价格下分布式存储提供的服务更好、性价比更高，且新节点的扩展以及坏旧节点的替换更为方便。

本书的目标读者

如果你是云存储方面的学者、工程师或架构师，那么本书适合你。

如果你是一位对云存储感兴趣的人或者是云存储的用户，那么光凭看这本书你可能无法实现一个自己的对象存储服务，但是在读完本书之后，你依然能够学到很多相关的知识。

对 Go 语言和 HTTP/REST 协议的了解可以帮助你实现并执行本书中的代码，但这不是必需的。本书对每一段代码都会有详细的解释来帮助读者了解其内容。即使对语言和协议一无所知的读者也能了解代码的含义和执行效果。

如果你是一位对云存储比较熟悉的读者，你可能已经了解对象存储服务的架构以及一些常见问题的成因，但这也不是阅读本书所必需的条件。本书会以提出问题并解决问题的方式介绍对象存储服务的架构设计。即便是对对象存储一窍不通的读者也可以在这个过程中亲眼见证对象存储服务的架构是如何一步步丰满起来的。

本书的主要内容

第 1 章，我们提出了一个单机版的对象存储原型系统，介绍了最简单的对象 REST 接口。

第 2 章，我们将这个原型系统进行了扩展，将它分拆成接口服务和数据服务，使得这些服务可以互相独立地提供服务功能，让我们的系统得以自由扩展。

第 3 章，我们又往系统中加入了元数据服务，用于保存描述对象的元数据，包括对象的名字、版本、大小、散列值等。有了元数据服务，我们就可以使得对象的名字和对象的内容解耦合。

第 4 章，我们实现对象数据的校验和去重，使得名字不同但内容相同的对象可以共享同一份存储实体，这样做可以降低对存储空间的要求。

第 5 章，为了增强数据的可靠性，我们提出了数据冗余的概念并实现了 RS 纠删码。我们在对象数据存取的过程中以流的形式进行编解码，可以在一定程度上修正对象数据的损坏。

第 6 章，为了战胜现实世界不良的网络环境，我们实现了断点续传。客户端在下载对象时自由指定下载数据的偏移量，也可以通过特殊的接口以分批的方式上传对象的数据。

第 7 章，我们介绍数据压缩。在大多数情况下，数据压缩都应该在客户端实现。但如果你需要设计一个通过浏览器就可以使用的对象存储系统，且你的大多数对象的数据都适合进行压缩，那么可以参考我们在本章实现的 gzip 数据压缩，进一步降低对存储空间和下载带宽的要求。

第 8 章，我们讨论了对象存储系统的数据维护，并实现了 3 个工具，它们可用于清理过期的对象数据和元数据，检查和修复当期的数据。

本书没有涉及的范畴

我们没有实现一个专门的客户端来配合对象存储系统，只是在部分章节中提到一个配套的客户端可以起到的作用。本书使用 Linux 下的 curl 命令作为我们的客户端进行功能测试，可以帮助我们更好地了解客户端和服务端之间发生的交互行为。但是一个美观 UI 的专门的客户端对用户来说会更加友好。

我们没有涉及用户管理，虽然用户管理是云存储系统的一个基本组成部分，但是这部分和其他系统的用户管理没什么区别，一个用户信息数据库就可以满足大多数要求，有兴趣的用户可以自行查阅相关书籍。

我们没有提到信息安全方面的内容，本书为了方便起见，使用的通信协议都是 HTTP，而事实上一个云存储系统对外一定是使用 HTTPS 协议，服务端和客户端之间需要建立 SSL 的双向认证。除此之外，用户合法身份的授权和验证等功能通常都会有一个专门的身份认证系统来进行管理，而服务端客户端则可以通过 JWT 和身份认证系统打交道。

我们没有实现对象存储系统的监控。系统监控包括对日志的实时收集和分析，对系统 KPI 的收集和可视化等。我们在这里推荐的做法是使用 Logstash 收集和分析系统日志和 KPI，记录在 ElasticSearch 中并用 Kibana 进行可视化。这些功能不涉及 Go 语言实现，而是通过各种工具的配置来进行。本书不展开讨论。

如何下载和运行本书中的代码

本书的代码使用 Go 语言实现，使用的 Go 编译器的版本是 1.8.1，开发和运行环境是 Ubuntu 16.04。

本书中所有 Go 语言代码实现都可以在 github 上找到，在 Linux 环境可以用 git 命令下载：

```
git clone https://github.com/stuarthu/go-implement-your-object-storage.git
```

github.com 是一个在线的软件项目管理仓库，Ubuntu 下的 git 客户端可以用 apt-get 下载安装：

```
sudo apt-get install git
```

编译 Go 代码需要运行 Go 编译器，读者可以在 Go 语言官网下载最新的 Go 编译器。

作者简介

胡世杰，上海交通大学硕士，目前任七牛云技术专家，是私有云存储服务的负责人。他是分布式对象存储系统专家，在该领域拥有多年的架构、开发和部署经验，精通 C/C++/Perl/Python/Ruby/Go 等多种编程语言，熟悉 ElasticSearch/RabbitMQ 等开源软件。

除了自己写作，他还致力于技术书籍的翻译，是《JavaScript 面向对象精要》《Python 和 HDF5 大数据应用》《Python 高性能编程》等著作的译者。

致谢

感谢我的妻子黄静和岳父黄雪春在我写书的日子里对我的支持，让我能够没有后顾之忧地写作。感谢人民邮电出版社的陈冀康编辑在本书写作和出版过程中的大力协助，感谢好友高博启发了我自己写作的念头，以及对本书推广所做的一切工作。

资源与支持

本书由异步社区出品，社区（https://www.epubit.com/）为您提供相关资源和后续服务。

配套资源

本书提供如下资源：

- 本书所有示例源代码；
- 本书作者针对书中内容的配套视频讲解。

读者请通过本书封底的刮刮卡观看。也可通过异步社区“课程”频道订阅。

要获得以上配套资源，请在异步社区本书页面中点击 配套资源，跳转到下载界面，按提示进行操作即可。注意：为保证购书读者的权益，该操作会给出相关提示，要求输入提取码进行验证。

提交勘误

作者和编辑尽最大努力来确保书中内容的准确性，但难免会存在疏漏。欢迎您将发现的问题反馈给我们，帮助我们提升图书的质量。

当您发现错误时，请登录异步社区，按书名搜索，进入本书页面，点击“提交勘误”，输入勘误信息，点击“提交”按钮即可。本书的作者和编辑会对您提交的勘误进行审核，确认并接受后，您将获赠异步社区的 100 积分。积分可用于在异步社区兑换优惠券、样书或奖品。

扫码关注本书

扫描下方二维码，您将会在异步社区微信服务号中看到本书信息及相关的服务提示。

与我们联系

我们的联系邮箱是 contact@epubit.com.cn。

如果您对本书有任何疑问或建议，请您发邮件给我们，并请在邮件标题中注明本书书名，以便我们更高效地做出反馈。

如果您有兴趣出版图书、录制教学视频，或者参与图书翻译、技术审校等工作，可以发邮件给我们；有意出版图书的作者也可以到异步社区在线提交投稿（直接访问 www.epubit.com/selfpublish/submission 即可）。

如果您是学校、培训机构或企业，想批量购买本书或异步社区出版的其他图书，也可以发邮件给我们。

如果您在网上发现有针对异步社区出品图书的各种形式的盗版行为，包括对图书全部或部分内容的非授权传播，请您将怀疑有侵权行为的链接发邮件给我们。您的这一举动是对作者权益的保护，也是我们持续为您提供有价值的内容的动力之源。

关于异步社区和异步图书

“异步社区”是人民邮电出版社旗下 IT 专业图书社区，致力于出版精品 IT 技术图书和相关学习产品，为作译者提供优质出版服务。异步社区创办于 2015 年 8 月，提供大量精品 IT 技术图书和电子书，以及高品质技术文章和视频课程。更多详情请访问异步社区官网 https://www.epubit.com。

“异步图书”是由异步社区编辑团队策划出版的精品 IT 专业图书的品牌，依托于人民邮电出版社近 30 年的计算机图书出版积累和专业编辑团队，相关图书在封面上印有异步图书的 LOGO。异步图书的出版领域包括软件开发、大数据、AI、测试、前端、网络技术等。

异步社区

微信服务号

分布式对象存储

第 1 章

目录

第1章

对象存储简介

1.1 和传统网络存储的区别

要理解对象存储，我们首先要来谈谈传统的网络存储。传统的网络存储主要有两类，分别是 NAS 和 SAN。

NAS 是 Network Attached Storage 的简称，是一个提供了存储功能和文件系统的网络服务器。客户端可以访问 NAS 上的文件系统，还可以上传和下载文件。NAS 客户端和服务端之间使用的协议有 SMB、NFS 以及 AFS 等网络文件系统协议。对于客户端来说，NAS 就是一个网络上的文件服务器。

SAN 是 Storage Area Network 的简称。它和 NAS 的区别是 SAN 只提供了块存储，而把文件系统的抽象交给客户端来管理。SAN 的客户端和服务端之间的协议有 Fibre Channel、iSCSI、ATA over Ethernet（AoE）和 HyperSCSI。对于客户端来说，SAN 就是一块磁盘，可以对其格式化、创建文件系统并挂载。

NAS 和 SAN 并不是完全对立的，现代的网络存储通常都是两者混合使用，可以同时提供文件级别的协议和块级别的协议。

介绍完传统的网络存储，那么对象存储跟它们又有什么区别呢？首先是对数据的管理方式不同。

1.1.1 数据的管理方式

对于网络文件系统来说，数据是以一个个文件的形式来管理的；对于块存储来说，数据是以数据块的形式来管理的，每个数据块有它自己的地址，但是没有额外的背景信息；对象存储则是以对象的方式来管理数据的，一个对象通常包含了 3 个部分：对象的数据、对象的元数据以及一个全局唯一的标识符（即对象的 ID）。

对象的数据就是该对象中存储的数据本身。一个对象可以用来保存大量无结构的数据，比如一首歌、一张照片或是一个在线文档。

对象的元数据是对象的描述信息，为了和对象的数据本身区分开来，我们称其为元数据。比如某首歌的歌名、某张照片拍摄的时间、某个文档的大小等都属于描述信息，也就是元数据。对于对象的元数据，我们在第 3 章会详细介绍，这里不多展开。

对象的标识符用于引用该对象。和对象的名字不同，标识符具有全局唯一性。名字不具有这个特性，例如张三家的猫名字叫阿黄，李四家的狗名字也可以叫阿黄，名字为阿黄的对象可以有很多个。但若是用标识符来引用就只可能有一个。通常我们会用对象的散列值来做其标识符，关于散列值的详细介绍见第 3 章。

除了对数据的管理方式不同以外，对象存储跟网络存储访问数据的方式也不同。

1.1.2 访问数据的方式

网络文件系统的客户端通过 NFS 等网络协议访问某个远程服务器上存储的文件。块存储的客户端通过数据块的地址访问 SAN 上的数据块。对象存储则通过 REST 网络服务访问对象。

REST 是 Representational State Transfer 的简称。REST 网络服务通过标准 HTTP 服务对网络资源提供一套预先定义的无状态操作。在万维网刚兴起的时候，网络资源被定义为可以通过 URL 访问的文档或文件。现如今对于它的定义已经更为宽泛和抽象：网络上一切可以通过任何方式被标识、命名、引用或处理的东西都是一种网络资源。

对于对象存储来说，对象当然就是一种网络资源，但除了对象本身以外，我们也需要提供一些其他的网络资源用来访问对象存储的各种功能，本书后续会一一介绍。

客户端向 REST 网络服务发起请求并接收响应，以确认网络资源发生了某种变化。HTTP 预定义的请求方法（Request Method）通常包括且不限于 GET、POST、PUT、DELETE 等，它们分别对应不同的处理方式：GET 方法在 REST 标准中通常用来获取某个网络资源；PUT 通常用于创建或替换某个网络资源（注意，它跟 PUT 的区别是 POST 一般不同于替换网络资源，如果该资源已经存在，POST 通常会返回一个错误而不是覆盖它）；POST 通常用于创建某个网络资源，DELETE 通常用于删除某个网络资源。

我们会在本书的后续章节看到对象存储的接口是如何使用这些HTTP请求方法的。

1.1.3　对象存储的优势

对象存储首先提升了存储系统的扩展性。当一个存储系统中保存的数据越来越多时，存储系统也需要同步扩展，然而由于存储架构的硬性限制，传统网络存储系统的管理开销会呈指数上升。而对象存储架构的扩展只需要添加新的存储节点就可以。

对象存储的另一大优势在于以更低的代价提供了数据冗余的能力。在分布式对象存储系统中一个或多个节点失效的情况下，对象依然可用，且大多数情况下客户都不会意识到有节点出了问题。传统网络存储对于数据冗余通常采用的方式是保留多个副本（一般至少 3 份，这样当其中一个副本出了错，我们还能用少数服从多数的方式解决争议），而对象存储的冗余效率则更高。我们会在第 5 章讨论数据冗余的问题。

本章将要实现的是一个单机版的对象存储原型，目的是让读者对我们讨论的对象存储有一个直观的了解。一个单机版的服务程序还称不上分布式服务，但是我们可以借此了解对象存储的接口，也就是说我们将了解客户端是如何通过 REST 接口上传和下载一个对象的，以及这个对象又是以什么样的形式被保存在服务器端的。从下一章开始，我们还将不断丰富架构和功能来适应各种新的需求。